ARMY RESEARCH LABORATORY

A Survey of Research in Supervisory Control and Data Acquisition (SCADA)

by Sidney C Smith

ARL-TR-7093 September 2014

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Aberdeen Proving Ground, MD 21005-5067

ARL-TR-7093 **September 2014**

A Survey of Research in Supervisory Control and Data Acquisition (SCADA)

Sidney C Smith

Computational and Informational Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
September 2014	Final	August 2013 - January 2014

4. TITLE AND SUBTITLE
A Survey of Research in Supervisory Control and Data Acquisition (SCADA)

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Sidney C Smith

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army Research Laboratory
ATTN: RDRL-CIN-S
Aberdeen Proving Ground, MD 21005-5069

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-7093

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
Author email: <sidney.c.smith24.civ@mail.mil>

14. ABSTRACT
That Supervisory Control and Data Acquisition (SCADA) systems are critical to the infrastructure of modern society and vulnerable to various attacks has been understood for several years. This report provides a brief overview of the technologies that compose SCADA systems, the impacts when SCADA systems fail, a high-level overview of the vulnerabilities present in SCADA systems, and a review of efforts that have been made to mitigate these vulnerabilities to include configuration guides, vulnerability assessments, network intrusion detection, cryptography, and test beds.

15. SUBJECT TERMS
SCADA, vulnerability, impact, countermeasures, configuration, assessment, network intrusion detection

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Sidney C Smith
Unclassified	Unclassified	Unclassified	UU	21	19b. TELEPHONE NUMBER (*Include area code*) 410-278-6235

Contents

List of Figures

INTENTIONALLY LEFT BLANK.

1. Introduction

Supervisory Control and Data Acquisition (SCADA) systems collect information from sensors and actuators about some physical environment and provide an operator interface which allows control and reporting. The physical environment could be a power plant, power-distribution network, water-treatment plant, manufacturing floor, petroleum refinery, or any other physical environment that requires control and data acquisition. As early as 1999 the European Organization for Nuclear Research was looking into using a SCADA system to control a laboratory environment.[1] As late as 2009 Aydogmus and Aydogmus discussed the use of a SCADA system to allow students to remotely interact with a laboratory supporting distance learning.[2] The data acquired may be operationally oriented and used to better run the system, or it could be strategic in nature and used to better run the company.

SCADA systems play a critical role in the foundation of modern society, and there is a large body of work dedicated to defending these systems from cyber attack. The purpose of this report is to survey and summerize that work. In Section 2, this report reviews the basic components of a SCADA architecture. In Section 3, it reviews the impacts of some famous failures in SCADA systems. In Section 4, this report discusses the shared and unique vulnerabilities of SCADA systems. In Section 5, it reviews the countermeasures that are currently in place to protect SCADA systems. In Section 6, the report provides some concluding insight into the current research being done to defend SCADA systems.

2. SCADA Architecture

Technically the SCADA system is composed of the information technology (IT) that provides the human-machine interface (HMI) and stores and analyzes the data. It may contain the logic necessary to operate the physical environment either autonomously or semi-autonomously. Although technically not part of the SCADA system, SCADA systems are connected to the sensors and actuators via a complex network of devices that include Front End Processors (FEPs), Intelligent Electronic Devices (IEDs), Master Terminal Units (MTUs), Motor Control Centers (MCCs), Programmable Logic Controllers (PLCs), and Remote Terminal Units (RTUs). FEPs provide a gateway from

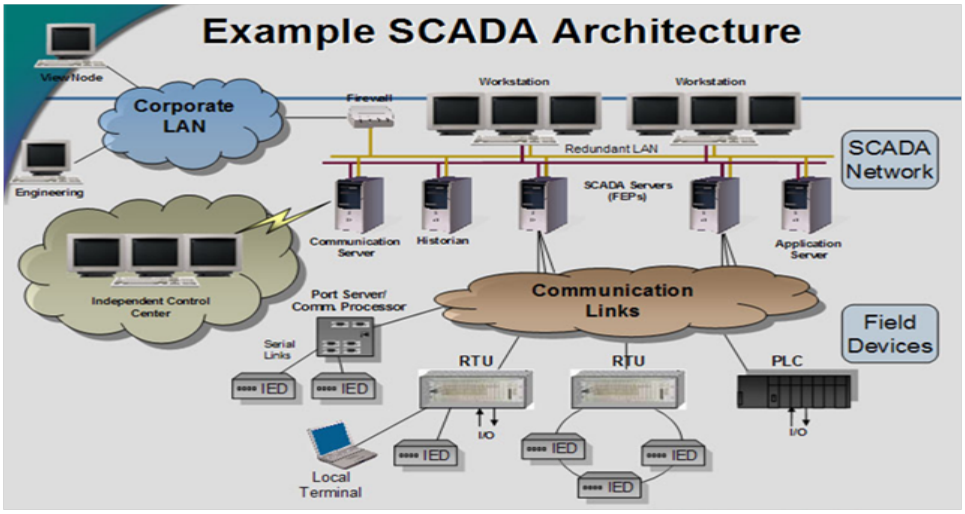

Figure An example of SCADA architecture (illustration by the US DoE[3])

the often proprietary protocols and media used by the sensor network to the general purpose computer system that runs the SCADA system. IEDs are very small, dedicated computer systems that directly connect to and control sensors and actuators. MTUs connect to multiple RTUs, consolidating the control and data traffic from the RTUs to the SCADA system. MCCs are dedicated computer systems that control the performance of an electric motor. PLCs provide a more direct interface between the sensor or actuator to the SCADA system encompassing the functionality of the IED and RTU into one device. An RTU connects to one or more IEDs and consolidates their data collection and control providing a gateway between the IED and the SCADA system or MTU.

The figure illustrates a typical SCADA architecture. Notice that the IEDs and PLCs connect directly to the sensors and actuators which are not shown in this diagram. The IEDs are connected to RTUs or an FEP, called a port server/comm processor in the diagram. These are then connected to the SCADA system which consists of a communications server, historian, and application server and services several workstations running the HMI.

Early SCADA implementations tended to be proprietary main-frame systems connected to a network of PLCs, IEDs, and MTUs, etc., with dedicated serial lines using proprietary protocols that were isolated from the corporate network. In the early 1990s this was shifting to an open systems and client server architecture on Reduced Instruction Set Computing (RISC) platforms running some version of UNIX.[4] Around the turn of the millennium work began on applying Web technologies to SCADA systems.[5–8] Lately the

trend has been to move from the UNIX/RISC system to commodity hardware and Microsoft solutions although there is some Linux,[1] to move from dedicated serial lines to Ethernet and even Wireless technologies, to move from proprietary protocols to standard protocols, and to connect these systems to the corporate network.[9] The proprietary HMIs have given way to Web-based applications. The requirement to share this information with customers and partners has spawned the implementation of Web-service-based applications. These migrations have greatly reduced the cost and increased the reliability and functionality of SCADA systems; however, they have also dramatically increased the risks inherent in them.

3. Impact of the Failure of SCADA Systems

SCADA systems control segments of the infrastructure critical to the smooth function of our society. These segments include the production and distribution of electricity, the distribution of water, the treatment of sewage, the refinement and distribution of petroleum—just to name a few.[10]

Fernandez and Fernandez[11] provide the following survey of the impact of SCADA failures that lead to 4 major blackouts in 2003:

> Outage 1: On August 25, 2003, more than 100 electric plants were shut down, including 22 nuclear power plants, affecting 50 million people in the U.S. and Canada. This was the biggest blackout in North American history, forcing the closure of 10 major airports, causing the cancellation of 700 flights, and leaving 350,000 people stranded on the New York City subway. A broken alarm at First Energy, a northern Ohio utility company, may have allowed too much to go wrong before technicians noticed the problem. The reversal of power happened so fast that operators did not have time to react, and within about 10 seconds, vast sections of the grid were overwhelmed. The failed lines in Ohio started a cascade that crashed several systems, despite a structure built for this type of defense.
>
> Outage 2: On August 29, 2003, a failure of England's National Electric Grid caused a blackout in Central and Southern London affecting more than 250,000 people, 270 sets of traffic lights, and 1,800 trains. According to the

latest findings, there was a fault in the volt system that apparently had not been properly maintained.

Outage 3: On September 23, 2003, Eastern Denmark and Southern Sweden experienced their worst blackout in 20 years. As maintenance issues have been raised, both of these outages have thus far been attributed to equipment failure.

Outage 4: On September 28, 2003, a power failure left most of Italy without power for several hours interrupting rail and air traffic and jamming emergency phone lines. Thousands were forced to take refuge in Rome's subways. As investigations revealed more information, it was found that the Italian response was either lacking, or too slow, and that Italian operators had made a wrong decision when coping with the interruption from Switzerland and France. Consequently, a cascade of power line outages resulted within Italy, and along its border.[11]

In 2011 Langer reported on a piece of malicious code that seems to target SCADA systems and the PLCs that they control.[12] This malicious code would come to be known as Stuxnet. Stuxnet was different from most other worms in that it appears to have limited its distribution to a targeted environment. It was initially introduced via a Universal Serial Bus (USB) device, and exploited four zero-day vulnerabilities in Microsoft Operations Systems to propagate itself throughout the SCADA system. It appears to have been looking for the systems used to download logic into particular Seimens PLCs. Once it verified that the PLCs were running the correct version of software, it would replace that software with its own malicious version. It appears that the intent of this code was to cause centrifuges in an uranium-enrichment facility to break down. Chen and Abu-Nimeh report, "The site's production dropped 15 percent in 2009, around the time Stuxnet is believed to have begun spreading."[13]

4. SCADA Vulnerabilities

SCADA systems share some common vulnerabilities in addition to some vulnerabilities that are quite unique. Modern SCADA systems are implemented on Commercial Off the Shelf (COTS) hardware and software solutions; therefore, they inherit all of the vulnerabilities associated with these systems.[11] Remember Stuxnet used zero-day

vulnerabilities in the Microsoft operating system to gain access to the PLCs where it installed its payload. More and more SCADA systems are using COTS networking technologies to include Ethernet and wireless networking, and they inherit all of the vulnerabilities inherent in them. The move to open communication standards has dramatically increased the availability of this information. Once upon a time the protocols used by SCADA systems to communicate with the sensors and actuators were proprietary and unavailable to most people. Today more than 90% of major SCADA and automation vendors have all of their manuals and specifications available on-line to the general public.[14] This is great for compatibility but not for security. Many of the control components such as PLCs, IEDs, and RTUs have very little additional computing power available to implement increased security features. COTS IT hardware and software have a life cycle of 3 to 5 years, but the life cycle of control components like PLCs, IEDs, and RTUs is 15 to 20 years. This implies that administrators of these systems are likely to be limited in their options because large pieces of their network are tied to technologies that would be considered woefully obsolete by IT standards.

5. SCADA Countermeasures

The fact that SCADA systems are very important and vulnerable has been understood,[15] and a significant body of work has been done to address these vulnerabilities. Many researchers and standards bodies have written configuration guides for SCADA systems. There are several researchers working vulnerability-assessment techniques. Work has been done in almost every area of intrusion detection. Research is being conducted in the use of cryptography and key management. Several people have created test beds for research into SCADA systems.

5.1 Configuration Guides

In 2001 Paul Oman et al., provided an overview of the risks to SCADA systems and provided 16 suggestions for hardening these systems.[16] In 2002 Jonathan Pollet advocated "Rings of Defense."[14] In 2005 Calvert Bowen et al., advocated "Defense in Depth" and client puzzles. They also introduce the concept of performing a Denial of Service (DoS) attack against oneself by implementing overly expensive countermeasures.[9]

In 2011 Arun Velagapalli et al., advocated minimizing the Trusted Computing Base for SCADA systems.[17] Many government and industry agencies have published comprehensive guidelines.[18]

There are limits to the effectiveness of configuration guides. If they are too complicated or expensive to implement, they will often be ignored unless some regulatory pressure exists to enforce them. In the IT field, configuration guides seem to suffer from a rapidly moving target. By the time a configuration guide can be written, tested, vetted, and approved the Operating System vendor has already released the next version requiring the configuration guide to be rewritten, tested, vetted, and approved. It is not at all unusual for systems in tightly regulated areas to be at least a version behind what is being currently offered. This means that brand new systems are being purchased with an operating system that is at least one version behind the latest. Given a life cycle of 3 to 5 years, organizations are often racing to upgrade before their systems reach their end of life. These configuration guides are often publicly available in order to ensure that the people who need them can get them; however, that means that we must assume that the adversary has them as well.

5.2 Vulnerability Assessments

Some very interesting work has been done to try to quantify the vulnerability of SCADA systems. In 2004 Eric Byres et al., used attack trees to analyze the vulnerabilities in the MODBUS/TCP SCADA protocol.[19] This work has direct application to protocol design and may make the next generation of protocols more secure. In 2006 Miles McQueen et al., used a directed compromise graph to determine the impact measured in time-to-compromise of security measures on a small SCADA system. He found an 86% reduction in vulnerabilities could yielded a 30% increase in time-to-compromise.[20] In 2007 Chee-Wooi Ten et al., proposed using attack trees to provide a quantitative vulnerability assessment of a power plant's SCADA system.[21] In 2008 he refined this approach; however, his models do not include OS Vulnerabilities.[22] All of these efforts are interesting proofs of concepts; however, their focus is so small as to bring into doubt whether their results would scale to real-world systems.

5.3 Network Intrusion Detection

There are basically 2 camps in network-intrusion detection: systems based upon signature detection and systems based upon anomaly detection. The signature-based systems look for traffic known to be malicious. The most popular and effective system

deployed to protect networks today are signature-based. Signature based can be very efficient and provide a low false-positive rate; however, they lack the ability to detect zero-day attacks. In other words, they are always behind the adversary because the signature cannot be written until someone has already been exploited.
Anomaly-detection systems work by learning what normal or benign traffic is and reporting on any abnormal traffic. These systems have the potential to detect zero-day exploits; however, it turns out that the complexity of modern networks makes discovering what normal or benign traffic is a very difficult problem. Historically these systems generate a significant number of false-positive alerts.

In the anomaly-detection camp are Steven Cheung et al., who in 2007 proposed using model-based intrusion detection specifically for SCADA networks that use MODBUS/TCP,[23] and Wei Gao et al., who in 2010 proposed using a neural-network-based, intrusion-detection system.[24] Cheung and Gao both assumed that the greater simplicity of SCADA systems and networks would allow them to get an accurate enough picture of normal or benign traffic that the number of false positives would be acceptable. They were able to present a proof of concept; however, their research was so tightly focused that it is not clear that their results would scale to real-world systems. Cheung focused upon one communication protocol out of the many that are in use in SCADA environments. Gao focused on only one attack vector out of the many that exist.

In 2007 Paul Oman and Matthew Phillips explored intrusion detection based upon pattern recognition.[15] Oman's system was more traditional signature-based with an anomaly-detection twist. He did do some very interesting work in the automated creation of his signatures. He was able to describe the correct operation of various devices and write rules that would fire if the traffic didn't match that pattern. The greatest limitation here is that rules would have to be written for every device in a SCADA network. This would be a very daunting task for any organization that had a primary mission other than research. It is this author's opinion that the only way this might be able to scale would be if vendors provided rule sets with their systems.

5.4 Cryptography

SCADA systems have the same authentication and confidentiality issues that are often addressed in other computer systems using cryptography. Encrypting the traffic should protect its confidentiality. The use of cryptographic signatures may be used to validate

authentication. One could envision securing the MODBUS/TCP and similiar protocols by adding a Secure Socket Layer much the same way that the Hyper Text Transport Protocotol was secured. In order to do this, crytographic modules would have to be installed on all devices for both Rivest Shamir Adleman (RSA) public key crytopgrphy and the Adanced Encyption Standard (AES) symmetric key cryptography. RSA is used for authentication and key exchange, and AES is used to encypt the session. That is a lot of software to load on these devices and a lot of computing to expect some of these devices to perform quickly enough to maintain their real-time requirements. An additional problem with all cyrptographic systems is the distribution of keys. This problem is exacerbated in a SCADA system because of the limited computing power, small bandwidth, and disconnected nature of the components. In 2006 Robert Dawson et al., proposed the SCADA Key Management Architecture as a replacement for SCADA Key Establishment.[25] In 2008 Ludovic Pietre-Cambacedes surveyed key management technologies for SCADA documenting Key-server, Point-to-point, Standard Public Key Infrastructure (PKI), and Customized PKI architectures.[26]

The long life cycle of this equipment presents a further challenge. A new protocol invented today may take 15 to 20 years to fully deploy. Additionally the strength of cryptography relies on certain problems being computationally infeasible. Moore's Law implies that what is computationally infeasible today will be feasible tomorrow. For example the Data Encyption Standard (DES) had a 56-bit key. When DES was developed, it was computationally infeasible to attempt every 56-bit combination to decode a message. Today brute-forcing a 56-bit DES key can be done very quickly. Given a 15- to 20-year life cycle, it is possible that the infeasibility assumptions will be invalid before the system is fully deployed.

5.5 Test Beds

Since most production SCADA systems are too valuable to use for research and most research organizations cannot afford to stand up a SCADA system, there has been a significant amount of work in creating SCADA test beds. Oman was able to use the University of Idaho's Electrical Engineering Power Laboratory, "a fully functioning high-voltage facility."[15] In 2008 Annarita Giani et al., described the SCADA test bed that they built to test DoS, Integrity, and phishing attacks.[27] In 2011 Thomas Morris et al., described the Mississippi State University SCADA security laboratory.[28]

6. Conclusions

SCADA systems are high-value targets where failure has the potential to cause massive amounts of damage. The life cycle of SCADA equipment means that some elements would be considered obsolete by computing standards. Configuration guides are good but will not be effective unless they are easy to understand and inexpensive to implement. Quantitative vulnerability-assessment techniques, although impressive, are of very narrow focus and based upon assumptions. Intrusion-detection work is significantly behind the same efforts in traditional network systems and what is being done is very highly focused on one particular technology. Older PLCs with low computing power make cryptography difficult to implement. Limited network connectivity makes key management difficult as well.

7. References

1. Daneels A, Salter W. What is scada. In: International Conference on Accelerator and Large Experimental Physics Control Systems; 1999 October, 4–8; Trieste, Italy. p. 339–343.

2. Aydogmus Z, Aydogmus O. A web-based remote access laboratory using scada. IEEE Transactions on Education 2009;52(1):126-132.

3. Department of Energy (US), Pacific Northwest National Laboratory. The role of authenticated communications for electric power distribution. Paper presented at: Network Embedded Control for Cyber Physical Systems (HCSS-NEC4CPS). 2006 November 8–9; Pittsburgh (PA).

4. Dy-Liacco T. Modern control centers and computer networking. IEEE Computer Applications in Power 1994;7(4):17-22.

5. Qiu B, Gooi HB. Web-based scada display systems (wsds) for access via internet. Power Systems, IEEE Transactions on 2000;15(2):681–686.

6. Ebata Y, Hayashi H, Hasegawa Y, Komatsu S, Suzuki K. Development of the intranet-based scada (supervisory control and data acquisition system) for power system. In: Power Engineering Society Winter Meeting, 2000. IEEE; Vol. 3; p. 1656–1661.

7. Li D, Serizawa Y, Kiuchi M. Concept design for a web-based supervisory control and data-acquisition (scada) system. In: Transmission and Distribution Conference and Exhibition 2002: Asia Pacific. IEEE/PES; Vol. 1; p. 32–36.

8. Qiu B, Gooi HB, Liu Y, Chan EK. Internet-based scada display system. Computer Applications in Power, IEEE 2002;15(1):14-19.

9. Bowen C, Buennemeyer T, Thomas R. Next generation scada security: best practices and client puzzles. In: Information Assurance Workshop, 2005. IAW '05. Proceedings from the Sixth Annual IEEE SMC; 2005 June 15–17; West Point (NY). p. 426–427.

10. Igure VM, Laughter SA, Williams RD. Security issues in scada networks. Computers & Security 2006;25(7):498 – 506.

11. Fernandez JD, Fernandez AE. Scada systems: vulnerabilities and remediation. J. Comput. Sci. Coll. 2005;20(4):160–168.

12. Langner R. Stuxnet: Dissecting a cyberwarfare weapon. Security Privacy, IEEE 2011;9(3):49-51.

13. Chen T, Abu-Nimeh S. Lessons from stuxnet. Computer 2011;44(4):91-93.

14. Pollet J. Developing a solid scada security strategy. In: Sensors for Industry Conference, 2002. 2nd ISA/IEEE; p. 148–156.

15. Oman P, Phillips M. Intrusion detection and event monitoring in SCADA networks. In: Critical Infrastructure Protection; Springer; 2007; p. 161–173.

16. Oman P, Schweitzer E, Roberts J. Safeguarding ieds, substations, and scada systems against electronic intrusions. In: Proceedings of the 2001 western power delivery automation conference; p. 9–12.

17. Velagapalli A, Ramkumar M. Minimizing the tcb for securing scada systems. In: Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research; CSIIRW '11, New York (NY): ACM; 2011. p. 19:1–19:1.

18. Ralston P, Graham J, Hieb J. Cyber security risk assessment for SCADA and DCS networks. ISA Transactions 2007;46(4):583 – 594.

19. Byres E, Franz M, Miller D. The use of attack trees in assessing vulnerabilities in scada systems. In: Proceedings of the International Infrastructure Survivability Workshop; 2004 December 5–8; Libson, Portugal. p. 1–9.

20. McQueen M, Boyer W, Flynn M, Beitel G. Quantitative cyber risk reduction estimation methodology for a small scada control system. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences HICSS '06; Vol. 9; 2006 January 4–7; Kauai (HI). p. 226–226.

21. Ten C-W, Liu C-C, Govindarasu M. Vulnerability assessment of cybersecurity for scada systems using attack trees. In: Power Engineering Society General Meeting, 2007. IEEE; p. 1–8.

22. Ten C-W, Liu C-C, Manimaran G. Vulnerability assessment of cybersecurity for scada systems. Power Systems, IEEE Transactions on 2008;23(4):1836-1846.

23. Cheung S, Dutertre B, Fong M, Lindqvist U, Skinner K, Valdes A. Using model-based intrusion detection for scada networks. In: Proceedings of the SCADA Security Scientific Symposium; 2007 January 24–25; Miami (FL). p. 1–12.

24. Gao W, Morris T, Reaves B, Richey D. On scada control system command and response injection and intrusion detection. In: eCrime Researchers Summit (eCrime), 2010; p. 1–9.

25. Dawson R, Boyd C, Dawson E, Nieto JMG. Skma: a key management architecture for scada systems. In: Proceedings of the 2006 Australasian workshops on Grid computing and e-research - Volume 54; ACSW Frontiers '06, Darlinghurst, Australia, Australia: Australian Computer Society, Inc.; 2006. p. 183–192.

26. Pietre-Cambacedes L, Sitbon P. Cryptographic key management for scada systems-issues and perspectives. In: International Conference on Information Security and Assurance, ISA; p. 156–161.

27. Giani A, Karsai G, Roosta T, Shah A, Sinopoli B, Wiley J. A testbed for secure and robust scada systems. SIGBED Review 2008;5(2):4.

28. Morris T, Vaughn R, Dandass YS. A testbed for scada control system cybersecurity research and pedagogy. In: Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research; 2011 ; New York (NY). CSIIRW '11, New York (NY): ACM; 2011. p. 1–4.

List of Symbols, Abbreviations, and Acronyms

AES	Advanced Encryption Standard
COTS	Commercial Off The Shelf
DES	Data Encryption Standard
DoS	Denial of Service
FEP	Front-End Processor
HMI	human-machine interface
IED	Intelligent Electronic Device
IT	information technology
MCC	Motor Control Center
MTU	Master Terminal Unit
OS	operating system
PKI	Public Key Infrastructure
PLC	Programmable Logic Controller
RISC	Reduced Instruction Set Computing
RSA	Rivest Shamir Adleman
RTU	Remote Terminal Unit
SCADA	Supervisory Control And Data Acquisition
USB	Universal Serial Bus

1 (PDF) DEFENSE TECHNICAL INFORMATION CTR DTIC OCA

2 (PDF) DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL IMAL HRA MAIL & RECORDS MGMT

1 (PDF) GOVT PRINTG OFC A MALHOTRA

1 (PDF) RDRL CIN S S SMITH